MITLL 2015 Language Recognition Evaluation System Description

Contributors in Alphabetical Order
Najim Dehak, Elizabeth Godoy, Douglas Reynolds*
Fred Richardson, Stephen Shum, Elliot Singer*
Doug Sturim, Pedro Torres-Carrasquillo

Human Language Technology Group
MIT Lincoln Laboratory
{dar, frichard, es, sturim, ptorres, elizabeth.godoy}@ll.mit.edu
*Spoken Language System Group
MIT-CSAIL
najim@csail.mit.edu, sshum@mit.edu

1 DEVELOPMENT DATA PREPARATION

1.1 SEGMENTATION OF THE FIXED (LIMITED) TRAINING CORPUS

The training data segments for the mandatory fixed training task made available by NIST was derived primarily from previously released data sources (Callhome, Callfriend, Mixer3, Switchboard, and VOA). **Error! Reference source not found.** shows a breakdown of the numbers of cuts and the speech duration (post-SAD) available for each language in the fixed training data set.

CODE	LANGUAGE	Cuts	Speech (hrs)	CODE	LANGUAGE	Cuts	Speech (hrs)
ara-acm	Iraqi	2206	75.59	por-brz	Braz. Port.	1838	5.96
ara-apc	Levantine	4073	266.67	qsl-pol	Polish	695	32.14
ara-arb	MSA	912	8.18	qsl-rus	Russian	2021	37.80
ara-ary	Maghrebi	919	46.91	spa-car	Carib. Spa.	194	30.59
ara-arz	Egyptian	440	97.27	spa-eur	Eur. Spa.	366	8.55
eng-gbr	British Eng.	147	2.10	spa-lac	Lat. Am. Spa.	160	15.30
eng-sas	Indian Eng.	1689	25.37	zho-cdo	Min	209	6.46
eng-usg	Amer. Eng.	2448	165.92	zho-cmn	Mandarin	4131	200.70
fre-hat	Hatian Cr.	2192	110.79	zho-wuu	Wu	234	10.36
fre-waf	West Afr. Fr.	1229	7.02	zho-yue	Cantonese	2382	123.61

Table 1: Cut and speech duration breakdown of data in the fixed training set.

In preparation for the 2015 NIST language identification evaluation the NIST data was separated into training and test sets to enable system development The 5,119 distributed files were separated by language into a 60% train set and a 40% test set. Because the durations of the files This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government..

range from 10 seconds to 30 minutes, the files were separated into sub-segments. SAD marks were used to extract multiple segments from the files such that resulting durations were uniformly distributed from 3 to 30 seconds. **Error! Reference source not found.** shows histograms of the durations of the sub-segments extracted from the training and test set. This expanded data set will give us the ability to calibrate the systems based on the duration seen in training and testing.

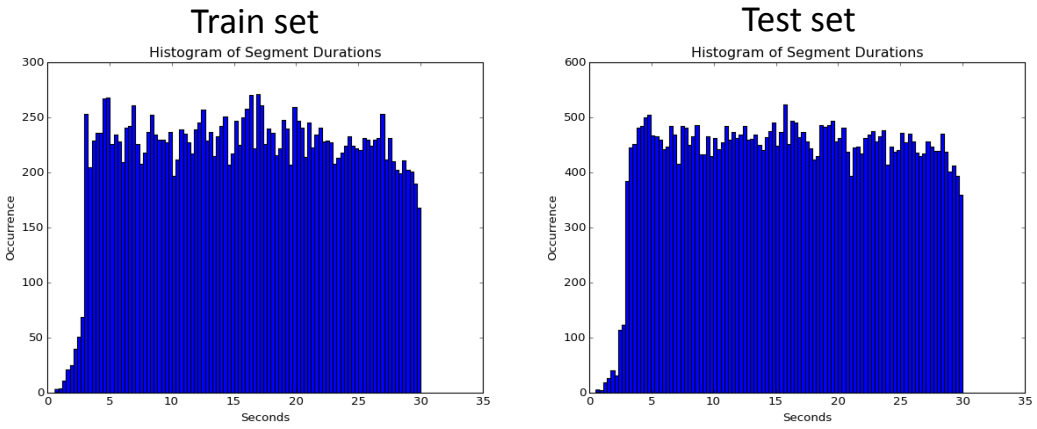

Figure 1: Histogram plots for the durations of the sub-segments for the train and test sets.

1.2 OPEN (UNLIMITED) DATA TRAINING CORPUS

Data for the unlimited data task was obtained by searching back through prior speech corpora for suitable segments. The labeled sources that were available to augment the data supplied by NIST are listed in Table 2: Labeled sources for training data that were available to augment the limited data training set. The amount of speech in each cut varied widely but is concentrated around 30 seconds. Table 3 shows a breakdown of the number of cuts and the speech duration (post-SAD) available for each language in the complete (NIST supplied plus augmented) unlimited training data set.

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

LANGUAGE	Sources	Type	Cuts
Arabic.egyptian	None available		
Arabic.iraqi	LRE11, Appen	CTS	1788
Arabic.levantine	LRE11, Fisher, Appen	CTS	3623
Arabic.maghrebi	LRE11	BNBS	505
Arabic.msa	LRE11	BNBS	506
Chinese.cantonese	LRE09, Babel	CTS, BNBS	2359
Chinese.mandarin	LRE05-07-09-11, Callfriend, OHSU	CTS, BNBS	3693
Chinese.minnan	LRE07-09	CTS	168
Chinese.wu	LRE07-09	CTS	189
Spanish.caribbean	LRE07	CTS	74
Spanish.european	Ahumada	CTS	328
Spanish.latinamerican	OHSU (Mexican)	CTS	130
Portuguese.brazilian	LRE09, OGI-22, VOA scrape	CTS, BNBS	1791
English.american	LRE05-07-09-11, Callfriend, OHSU	CTS	2088
English.indian	LRE07-09-11, OHSU, OGI-22	CTS	1271
English.british	UK-MI5 SID	CTS	148
Polish	LRE11	CTS, BNBS	208
Russian	LRE07-09-11, Callfriend	CTS, BNBS	1551
West African French	LRE09, VOA scrape	BNBS	1195
Haitian Creole	Babel, VOA scrape	CTS, BNBS	1869

Table 2: Labeled sources for training data that were available to augment the limited data training set.

CODE	LANGUAGE	Cuts	Speech (hrs)	CODE	LANGUAGE	Cuts	Speech (hrs)
ara-acm	Iraqi	2206	75.59	por-brz	Braz. Port.	1838	5.96
ara-apc	Levantine	4073	266.67	qsl-pol	Polish	695	32.14
ara-arb	MSA	912	8.18	qsl-rus	Russian	2021	37.80
ara-ary	Maghrebi	919	46.91	spa-car	Carib. Spa.	194	30.59
ara-arz	Egyptian	440	97.27	spa-eur	Eur. Spa.	366	8.55
eng-gbr	British Eng.	147	2.10	spa-lac	Lat. Am. Spa.	160	15.30
eng-sas	Indian Eng.	1689	25.37	zho-cdo	Min	209	6.46
eng-usg	Amer. Eng.	2448	165.92	zho-cmn	Mandarin	4131	200.70
fre-hat	Hatian Cr.	2192	110.79	zho-wuu	Wu	234	10.36
fre-waf	West Afr. Fr.	1229	7.02	zho-yue	Cantonese	2382	123.61

Table 3: Cut and speech duration breakdown of all data available for the unlimited training set.

During development testing it was found that using all the extra data to augment the training data provided by NIST hurt language recognition performance for some clusters. Therefore, another set of runs was performed in which the limited set was augmented by the extra data from each of the languages in turn. It was found that only 3 of the languages contributed to improved

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

performance: Brazilian Portuguese, British English, and Arabic MSA. Consequently, reported results for the unlimited task included extra data from these languages only.

2 SYSTEMS

A total of 6 systems were developed at MIT-CSAIL and 5 systems at MIT Lincoln Laboratory (MIT-LL).

MIT-CSAIL Systems

BNF1: CSAIL bottleneck i-vector system
CNT1: Multinomial i-vector system trained with ASR senone posteriors
CNT2: Multinomial i-vector system trained with language class targets posteriors
CNT3: Multinomial i-vector system trained with both ASR senone and language class targets
BAUD: Unsupervised DNN BNF systems

MIT-LL Systems

MMI: MMI trained Gaussian classifier using the BNF features
IVEC: i-vector classifier trained using SDC features
BNF2: i-vector classifier trained using the BNF features
STATS: i-vector classifier trained using the DNN posteriors and SDC features
PITCH1: i-vector classifier trained using SDC and pitch features
PITCH2: i-vector classifier trained using BNF and pitch features

2.1 CSAIL BOTTLENECK I-VECTOR SYSTEM (BNF1)

The Deep Neural Network architecture that we used for this system was composed of seven hidden layers. The sixth layer was based on linear activation nodes with a dimension of 80. This hidden layer was used to extract the bottleneck features. The rest of the hidden layers used sigmoid activation with 1024 neurons. This DNN was trained on 90% of the Switchboard phase 1 dataset, with the 10% remaining data used as the development set. We used the Kaldi toolkit to extract 4168 senones posteriors. The trained DNN was based on 21-stacked PLP frames of dimension 13 and both first and second derivatives [Richardson2015].

The bottleneck feature vectors were then normalized to a standard normal distribution for each file. These features were used to train the GMM-UBM and I-vector models. A UBM comprising 2048 Gaussians was trained on the training dataset. The i-vector of dimension 400 [Dehak2011] was trained on the training data as well as on augmented audio data transformed using both the speed (0.9,1.1) and tempo (0.9,1.1) options of sox toolkit [Ko2015]. In general the data was augmented by a factor of 4. This augmented data and short cuts of different durations (3s, 10s, 30s) extracted from the same dataset were used to train Linear Discriminant Analysis (LDA), Within Class Covariance Normalization (WCCN) and the mean for each class. We applied cosine scoring to compute to compute the decision score [Singer2012]. This scoring is a simplified version of the Von Mises Fisher distribution.

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

2.2 ASR COUNTS SUBSPACE SYSTEM (CNT1)

We trained a DNN using the same setup as the one used to extract bottleneck features. This DNN was characterized by 7 hidden layers of dimensions 2048:1024:2048:1024:2048:1024:2048 and 4168 posterior outputs. It was also based on the same 21-stacked PLP frames. This DNN was used to extract posterior statistics of all hidden layers. The Subspace Multinomial Model was then applied to model these zero order statistics. We trained an 800-dimension subspace on the long and short cuts of the original audio but without the speed/tempo manipulated cuts. WCCN and the mean of each class were trained in the short cuts of the augmented data similar to the bottleneck system. Cosine scoring was applied to compute the final decision score.

2.3 LID COUNTS SUBSPACE SYSTEM (CNT2)

This system is very similar to the previous one (ASR counts subspace system) except that we used a DNN with stacked PLP features as input and the language class as output. This DNN was composed of 7 hidden layers of dimensions 2048:1024:2048:1024:2048:1024:2048 and 20 posterior outputs representing the language classes. Using the DNN, we extracted the posterior statistics from all hidden layers. These statistics were then modeled with a Subspace Multinomial Model of dimension 800. This subspace was trained on the long and short cuts of the original audio but without the speed/tempo manipulated cuts. We used the same cosine scoring as previous systems.

The **CNT3** system is the combination of **CNT1** and **CNT2**.

2.4 BAUD SYSTEM DESCRIPTION (BAUD)

This bottleneck feature-based system is similar in concept to the one proposed in [Richardson2015], but instead of training the DNN using senone targets from the tri4a step of the Kaldi SWB recipe, this system trained its bottleneck features using targets from an unsupervised unit discovery process, which we detail below. Aside from this new set of targets, the framework of this system follows the exact same format as [Richardson2015]. Specifically, the DNN stacks ±10 frames of 13-dimensional PLP features with their first- and second-order derivatives as input (i.e., (10+1+10) * (13+13+13) = 819 dimensions) to a 7-layer DNN with a 64-unit linear bottleneck at the second-to-last (i.e., 6th) layer. All other hidden layers contain 1024 units with sigmoid activations.

We extract bottleneck features from the provided training data to build our 2048-Gaussian UBM. To train our 600-dimensional i-vector extractor, we used the UBM data as well as augmented audio data transformed using the sox toolkit, which varies both speed and tempo at rates of 0.9x and 1.1x [Ko2015]. This increases the amount of data by a factor of four. Finally, i-vectors from the training data, the speed/tempo-augmented data, and a set of shorter cuts of various durations (3s, 10s, 30s) extracted from the same training data were all used to train a Linear Discriminant Analysis (LDA) transform and a Within-Class Covariance Normalization (WCCN). After applying all these transformations, we compute a mean for each language class and obtain decision scores for each test i-vector via cosine scoring, which can be seen as a simplified

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

version of the von Mises-Fisher distribution [Singer2012]. Note that these steps are equivalent to that of the CSAIL bottleneck i-vector system; the only difference being in the bottleneck features themselves.

The unsupervised unit discovery process (also known as Bayesian acoustic unit discovery, or BAUD) is based off the work in [Lee2012], but was subsequently re-implemented in Kaldi with a few simplifications to make the computation more tractable [Harwath2015]. The main idea is to learn phone-like units on speech without parallel text data. Each unit is represented by a 3-state HMM that emits acoustic feature vectors via a GMM. In [Lee2012], everything was formulated in a Bayesian manner to take advantage of its self-regularizing model-selection properties, and inference was done via Gibbs sampling. In the faster re-implementation, we used a more heuristic initialization, which included specifying the number of units to learn, and accumulated GMM statistics via maximum likelihood.

We learned 100 units on all of the provided training data. This resulted in a large set of "phone sequences" from which we could train a speech recognizer in Kaldi. Carrying through to the tri2 step of the SWB recipe resulted in an acoustic model containing 2604 senones modeled using 30,000 Gaussians. The frame-level alignments for these senones were used to train the DNN for bottleneck feature extraction.

2.5 I-VECTOR CLASSIFIER (IVEC)

The IVEC system used Shifted Delta Cepstra features as input. The SDC features used speech windowing of 20 ms length and 10 ms shift. Window DC was subtracted and a low energy dither was added to the windowed speech to avoid digital zeros. RASTA filtering of log-energy filterbank sequences was applied. SDC features are extracted using parameters d=1, p=3, k=7, and static cepstra were prepended to produce a 56 dimensional feature vector.

The i-vector classifier used a 2048 component GMM and 600 dimensional i-vector subspace. The total covariance matrix was trained using the EM algorithm. The system used simple cosine scoring because it produced performance superior to that of WCCN and PLDA scoring in our initial experiments.

2.6 I-VECTOR CLASSIFIER TRAINED USING BNF FEATURES (BNF2)

A DNN classifier for the **BNF2** system was trained over a 100 hour subset of the Switchboard 1 training set. The subset was determined by the Kaldi training recipe for Switchboard. The DNN consisted of 7 layers of 1024 nodes each with a 64 node linear bottleneck at the second to last layer. The DNN was trained using 4199 output classes also determined by the Kaldi training recipe. The input to the DNN consisted of 39 PLP features (which include Δ and $\Delta\Delta$ coefficients) warped to fit a standard Gaussian distribution over a 300 frame window and stacked using a symmetric ± 10 frame window. Thus the input to the DNN consisted of 819 features (39 features over 21 stacked frames). More details can be found in [Richardson2015].

The **BNF2** systems features were then fed into the i-vector classifier described in Section 2.5.

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

2.7 AN MMI TRAINED GAUSSIAN CLASSIFIER USING THE BNF FEATURES (MMI)

The **MMI** system is a trained Gaussian classifier [Matejka2007] using the BNF features of system **BNF2**.

2.8 THE I-VECTOR CLASSIFIER TRAINED USING THE DNN POSTERIORS AND SDC FEATURES (STATS)

This i-vector based system used the 4199 DNN posteriors and the 56 SDC features to form supervectors. The rest of the i-vector system is the same (EM training is used to estimate the T-matrix and the sub-space dimension for the i-vectors is 600) [Lei2014, Richardson2015]. The core i-vector classifier is described above (**IVEC**).

2.9 SDC, BNF, AND PITCH I-VECTOR SYSTEM (PITCH1, PITCH2)

The pitch based systems used pitch stacked with the SDC features of the **IVEC** system (**PITCH1**) and the bottleneck features of the **BNF2** system (**PITCH2**). Pitch features were generated on a per-cut basis. Praat [Boersma2013] was used to calculate F_0 and the corresponding voicing decision. This was done at 10 millisecond frame rate, with the F_0 range set between a minimum of 65 Hz and a maximum of 400 Hz. To mitigate the effects of pitch doubling and pitch having, the highest and lowest 3% of F_0 values were removed. The log of F_0 was taken and its mean over the voiced frames of the cut was subtracted. Linear interpolation of the log(F_0) measure was performed through the unvoiced frames and those with the most extreme F_0 values were removed. Delta-log(F_0) was calculated as the difference between the log(F_0) value 3 frames forward and 3 frames back in time. The values of log(F_0) and delta-log(F_0) were stacked with the corresponding SDC frames, producing a new 58 dimensional feature vector for the **PITCH1** system. The **PITCH2** system used values of log(F_0) and delta-log(F_0) stacked with the **BNF2** systems features.

The i-vector system **IVEC** described in Section 2.5 was used as a classifier for both pitch-based systems.

3 FUSION FOR THE FIXED (LIMITED) CONDITION

Prior to system fusion, each system was calibrated using an MMI trained tied covariance Gaussian backend with duration normalization [McCree2008, Singer2012]. The input features and the output scores for the calibration subsystem corresponded to each of the 20 languages classes. Fusion used a simple logistic regression with a single weight for each system [Singer2012]. The candidates for fusion included the 6 systems developed at MIT Lincoln Laboratory (MIT-LL) and the 5 systems developed at MIT CSAIL (CSAIL), as described in Section 2.

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

The top systems for fusion were selected by sweeping across all system combinations using scores on our test partition. The backend for the final evaluation submission was trained over scores on the entire fixed data training corpus (including both our training and test partitions).

WCCN versions of the **BNF1** and **CNT1-3** systems with and without LDA were evaluated for fusion as well as an additional cosine (**COS**) scoring version of the BNF1 system bringing the total number of CSAIL system outputs to 10. Table 4 summarizes the MIT-CSAIL scoring sub-system types used for fusion.

System	WCCN	LDA+WCCN	Cosine
BNF1	✓	✓	✓
CNT1	✓	✓	
CNT2	✓	✓	
CNT3	✓	✓	
BAUD		✓	

Table 4: MIT-CSAIL scoring type sub-systems

3.1 FUSION SWEEP

Given the difficulty in evaluating all possible fusions of 16 system outputs we adopted a three stage strategy for system selection:

Stage 1 sweep:

The first stage of system selection included only the outputs of the 6 MIT-LL systems. From this sweep we kept all systems except for **MMI**.

Stage 2 sweep:

Next, the 5 remaining MIT-LL systems together with all three scoring type versions of the **BNF1** system were combined, and from the resulting 8-way system fusion analysis the **BNF1**, **STATS/COS,** and **PITCH1/COS** systems were retained and the **IVEC/COS**, **BNF2/COS,** and **PITCH2/COS** systems were dropped.

Stage 3 sweep:

Finally, all 10 CSAIL system in together with the remaining MIT-LL **STATS** and **PITCH1** systems were evaluated. From an analysis of all possible 12-way combinations we chose to keep the 5 systems listed in Table 5 for our primary fixed-task submission.

System
BAUD/LDA+WCCN
CNT1/WCCN
BNF1/LDA+WCCN
PITCH1/COS

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

STATS/COS

Table 5: Final system fusion for primary fixed submission.

We also selected the single best system (**BNF1/LDA+WCCN**) as a secondary submission.

3.2 PER-CLUSTER FUSION

Another secondary system was submitted by finding the top ranking fusion for each language cluster in the Stage 3 fusion sweep described above. The systems used for each language cluster are shown in Table 6.

Language cluster	Systems
Arabic	**BAUD, CNT1/LDA+WCCN, BNF1/WCCN, STATS/COS**
Chinese	**BAUD, CNT1/WCCN, BNF1/LDA+WCCN, STATS/COS**
English	**CNT1/LDA+WCCN, BNF1/COS, BNF1/WCCN, STATS/COS**
French	**CNT3/WCCN, BNF1/LDA+WCCN**
Iberian	**IVEC, BNF1/COS, PITCH1/COS, PITCH2/COS**
Slavic	**BNF1/COS, PITCH2/COS, STATS/COS**

Table 6: Per-language cluster system fusion.

4 FUSION FOR THE OPEN (UNLIMITED) TRAINING CONDITION

Multi-lingual DNN: Inspired by the work described in [Fer2015], a multi-task DNN was trained using data from 5 IARPA Babel languages described in Table 7. A DNN was trained using 60 hours of data randomly selected from each language for a total of 300 hours of data. The inputs for the DNN were the same stacked features used for the **BNF2** system. The DNN architecture is also similar to the **BNF2** system in that it has 7 layers of 1024 nodes each and the second to last layer is a 64 node linear bottleneck. However for the multi-lingual DNN the last hidden layer is different for each of the five languages. Stochastic gradient descent training for the multi-lingual DNN proceeds by loading a mini-batch with data from each language in sequence until the average validation cost across all languages no longer decreases.

Language	IARPA Build Pack
Cantonese	IARPA-babel101b-v0.4c
Pashto	IARPA-babel104b-v0.bY
Turkish	IARPA-babel105b-v0.4
Tagalog	IARPA-babel106b-v0.2g
Vietnamese	IARPA-babel107b-v0.7

Table 7: 5 Babel languages used for training a multi-lingual BNF.

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

In addition to the 5 systems used in the limited training primary system submission, 5 more systems were trained on language recognition data that was not part of the fixed LRE15 data set. These systems are listed in Table 8.

System	Description
BNF2UC	The **BNF2** system trained on the open data set.
IVECUC	The **IVEC** system trained on the open data set.
STATSUC	The **STATS** system trained on the open data set.
PITCH2UC	The **PITCH2** system trained on the open data set.
MLBNFUC	An i-vector system trained using bottleneck features from the multi-lingual DNN described above.

Table 8: Systems used for the open submission fusion sweeps.

Both WCCN and cosine scoring outputs of these systems were used in evaluating possible system combinations for system fusion. This amounts to a total of 15 possible systems for fusion: the 10 systems in Table 8 for both WCCN and cosine scoring together with the 5 systems used for the primary limited submission

4.1 FUSION SWEEP

As with the fixed primary system fusion sweeps described above, given the difficulty in evaluating all possible fusions of 15 system outputs for the open submission, we adopted a three stage strategy.

Stage 1 sweep:

The first stage of system fusion sweeps evaluated all combinations of the 10 outputs (WCCN and cosine) from the 5 systems in Table 8. Analysis of the results for these 10-way combinations led to selecting the following three systems: **STATSUC/COS**, **PITCH2UC/WCCN,** and **MLBNFUC/COS.**

Stage 2 sweep:

The final system fusion sweep included 5 systems used in the final fixed training condition submission and the three systems selected in the previous stage for a total of 8 systems. From the sweep of all possible system combinations, the systems listed in Table 9 were selected for the primary open training system fusion submission.

System
STATSUC/COS
MLBNFUC/COS
CNT1/WCCN

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

BNF1/LDA+WCCN
PITCH1/COS

Table 9: Final system fusion used for the open system submission.

Two other systems were submitted for the open training condition: the single-best **MLBNFUC/COS** system and the top scoring 3-system combination consisting of the **MLBNFUC/COS, BNF1/LDA+WCCN,** and **PITCH1/COS** systems.

5 REFERENCES

[Boersma2013] Paul Boersma & David Weenink (2013): Praat: doing phonetics by computer [Computer program]. Version 5.3.51, retrieved 2 June 2013 from http://www.praat.org

[Dehak2011] N. Dehak, P. Torres-Carrasquillo, D. Reynolds, and R. Dehak, "Language Recognition via Ivectors and Dimensionality Reduction," Proc. Interspeech, pp. 857-860, Florence, Italy, August, 2011.

[Fer2015] R. Fer, P. Matejka, F. Grezl, O. Plchot and J. Cernock, "Multilingual Bottleneck Features for Language Recognition," Proc. Interspeech, 2015.

[Harwath2015] Thanks to David Harwath (SLS, MIT CSAIL) for providing the code and support of his re-implementation of BAUD in Kaldi.

[Ko2015] Tom Ko, Vijayaditya Peddinti, Daniel Povey and Sanjeev Khudanpur, "Audio Augmentation for Speech Recognition", Proc. Interspeech, 2015.

[Lee2012] C. Lee and J. Glass, "A Nonparametric Bayesian Approach to Acoustic Model Discovery," Proceedings of ACL, July 2012.

[Matejka2007] Matejka, Pavel, et al. "BUT system description for NIST LRE 2007." Proc. 2007 NIST Language Recognition Evaluation Workshop, 2007.

[McCree2008] A. McCree, F. Richardson, E. Singer, D. A. Reynolds, "Beyond frame independence: parametric modelling of time duration in speaker and language recognition," Proc. Interspeech, 2008

[Lei2014] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in Proc. of ICASSP 2014

[Richardson2015] F. Richardson, D. Reynolds, and N. Dehak, "Deep Neural Network Approaches to Speaker and Language Recognition," *IEEE Signal Processing Letters*, October 2015, Vol. 22, No. 10, pp. 1671-1675.

[Singer2012] E. Singer, P. Torres-Carrasquillo, D. Reynolds, A. McCree, F. Richardson, N.

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

Dehak, and D. Sturim, "The MITLL NIST LRE 2011 Language Recognition System," Proc. Odyssey, pp. 209-215, Singapore, June 2012.

This work was sponsored by the Department of Defense under Air Force contract F19628-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.